

## CLAIMS

### **What is claimed is:**

- 5     1. A method in a data processing system, the method comprising the steps of:  
         receiving a request to generate a distributed computing component;  
         generating code corresponding to the distributed computing component, the code  
         containing a method having one of a plurality of types; and  
         displaying a graphical representation of the code that includes a separately  
10     delineated display area for each of the plurality of types.
2. The method of claim 1, further comprising the step of displaying a symbol in each  
         separately delineated display area, wherein the symbol indicates the type of method  
         displayed in the display area.
3. The method of claim 1, wherein the graphical representation of the code includes a  
15     separately delineated display area for reference types in the code.
4. The method of claim 1, wherein the one type is a create method type.
5. The method of claim 1, wherein the one type is a finder method type.
6. The method of claim 1, wherein the one type is a business method type.
7. The method of claim 1, wherein the distributed computing component is an  
20     Enterprise JavaBean™.
8. The method of claim 1, wherein the distributed computing component is a  
         SessionBean.
9. The method of claim 1, wherein the distributed computing component is an  
         EntityBean.

10. A method in a data processing system, the method comprising the steps of:  
receiving a request to generate a distributed computing component;  
generating code corresponding to the distributed computing component, the code  
containing a method having one of a plurality of types;  
5 associating a symbol with each type; and  
displaying a graphical representation of the method with the symbol indicating  
the type of the method.

11. The method of claim 10, wherein the step of generating code comprises the steps of:  
generating code corresponding to an implementation class;  
10 generating code corresponding to a home interface that is related to the  
implementation class; and  
generating code corresponding to a remote interface that is related to the  
implementation class.

12. The method of claim 11, wherein the code corresponding to the implementation class  
15 contains the method and the code corresponding to the home interface contains a  
signature of the method.

13. The method of claim 12, further comprising the step of displaying the symbol  
indicating the type of the method with a graphical representation of the signature.

14. The method of claim 11, wherein the code corresponding to the implementation class  
20 contains the method and the code corresponding to the remote interface contains a  
signature of the method.

15. The method of claim 14, further comprising the step of displaying the symbol  
indicating the type of the method with a graphical representation of the signature.

16. The method of claim 11, further comprising the step of displaying a graphical  
25 representation of the code corresponding to the implementation class that includes a first  
separately delineated display area for a create method type, a second separately  
delineated display area for a finder method type, and a third separately delineated display  
area for a business method type.



19. A method in a data processing system comprising a software development tool, the method comprising the steps of:

initiating execution of the software development tool; and

while the software development tool is executing, the software development tool

5 performs the steps of:

receiving an indication to generate a distributed computing component;

generating source code corresponding to the distributed computing component;

compiling the source code to form executable code for the distributed computing component;

10 generating a deployment descriptor file that includes a plurality of properties to control deployment of the distributed computing component on a computer and to control a relationship between the distributed computing component and a deployment environment on the computer;

15 storing executable code for the distributed computing component and the deployment descriptor file in a deployment archive;

deploying the distributed computing component to the computer; and

initiating execution of the distributed computing component on the computer.

20. The method of claim 19, wherein the source code includes an implementation class.

21. The method of claim 19, wherein the source code includes a home interface.

20 22. The method of claim 19, wherein the source code includes a remote interface.

23. The method of claim 19, wherein the source code includes a primary key class.

24. The method of claim 19 further comprising the step of storing deployment information associated with the distributed computing component in a comment of the source code for the distributed computing component.

25 25. The method of claim 24, wherein the step of generating the deployment descriptor file further comprises the steps of:

retrieving the deployment information associated with the distributed computing component from the comment of the source code corresponding to the distributed computing component; and

30 storing the deployment information in the deployment descriptor file.

26. The method of claim 25, further comprising the steps of:

receiving a change to the deployment information associated with the distributed computing component contained in the deployment descriptor file; and

35 modifying the deployment information in the comment of the source code for the distributed computing component to reflect the change.

27. The method of claim 24, wherein the deployment information includes at least one of the plurality of properties.

28. The method of claim 19, wherein the step of deploying the distributed computing component comprises the step of transferring the deployment archive to the computer.

40 29. The method of claim 19, wherein the software development tool is located on the computer.

30. The method of claim 19, further comprising the step of initiating execution of a debugger to facilitate testing the distributed computing component on the computer.

45 31. The method of claim 30, wherein the step of initiating the execution of the distributed computing component comprises initiating the execution of the distributed computing component in one debug session.

32. The method of claim 30, further comprising the steps of:

receiving an indication of a client software component; and

initiating execution of the client software component in another debug session.

50 33. The method of claim 32, wherein the client software component when executing invokes a method associated with the distributed computing component.

34. The method of claim 19, further comprising the steps of:

generating a plurality of web pages to test the distributed computing component after deployment; and

55 providing a client application with an address for accessing the web pages.

35. The method of claim 34, wherein one of web pages contains code that is responsive to an invocation from the client application to execute a method in the executable code of the distributed computing component.

36. The method of claim 35, wherein another of the web pages contains code that  
60 provides a result to the client application in response to the execution of the method in the executable code of the distributed computing component.

14144582/v2

37. A method in a data processing system comprising a software development tool, the method comprising the steps of:
  - initiating the execution of the software development tool; and
  - while the software development tool is executing, the software development tool
- 5 performing the steps of:
  - receiving an indication to deploy a distributed computing component;
  - deploying the distributed computing component to a deployment environment on a computer; and
  - initiating execution of the distributed computing component in debug mode.
- 10 38. The method of claim 37, further comprising the steps of:
  - receiving a debug command; and
  - in response to receiving the debug command, controlling the execution of the distributed computing component while in debug mode.
39. The method of claim 38, further comprising the step of, in response to receiving the
- 15 debug command, displaying a result from executing the distributed computing component.
40. The method of claim 37, wherein the step of initiating the execution of the distributed computing component further comprises automatically initiating execution of the distributed computing component in one debug session.
- 20 41. The method of claim 40, further comprising the steps of:
  - receiving an indication of a client software component; and
  - automatically initiating execution of the client software component in another debug session.

42. The method of claim 41, further comprising the steps of:

25       receiving a first debug command;

          receiving a second debug command;

          in response to receiving the first debug command, controlling the execution of the

client software component in the other debug session; and

          in response to receiving the second debug command, controlling the execution of

30       the distributed computing component in the one debug session.

43. The method of claim 42, further comprising the steps of:

          in response to receiving the first debug command, displaying a first result from

executing the distributed computing component; and

          in response to receiving the second debug command, displaying a second result

35       from executing the client software component.

44. The method of claim 43, wherein the client software component when executing

invokes a method associated with the distributed computing component.

45. The method of claim 44, wherein the first result and the second result reflect the

invocation of the method.

40



46. A method in a data processing system comprising a software development tool with access to a distributed computing component, the method comprising the steps of:

initiating the execution of the software development tool; and

while the software development tool is executing, the software development tool

5 performing the steps of:

receiving an indication to deploy the distributed computing component; and

generating a web page to facilitate testing the distributed computing component.

47. The method of claim 46, further comprising providing a client with an address for accessing the web page to test the distributed computing component.

10 48. The method of claim 47, wherein the web page has code that is responsive to an invocation from the client to execute a method in executable code of the distributed computing component.

49. The method of claim 48, wherein the code of the web page provides a result to the client in response to the execution of the method in the executable code of the distributed  
15 computing component.

50. The method of claim 48, wherein the method corresponds to a create method.

51. The method of claim 48, wherein the method corresponds to a finder method.

52. The method of claim 48, wherein the method corresponds to a business method.

53. A method in a data processing system having a distributed computing component with code, the distributed computing component having a type, the method comprising the steps of:

5 determining whether the code of the distributed computing component has a non-compliant portion that does not comply with a specification for the type of distributed computing component; and

when it is determined that the code corresponding to the distributed computing component has a non-compliant portion, replacing the non-compliant portion with new code that complies with the specification.

10 54. The method of claim 53, further comprising the step of, after replacing the non-compliant portion with new code, refactoring the code corresponding to the distributed computing to modify a reference to the non-complaint portion.

15 55. The method of claim 53, wherein the specification is associated with a target application server that is capable of controlling execution of the distributed computing component.

56. The method of claim 53, wherein the specification corresponds to one of a plurality of Enterprise JavaBean™ specifications.

20 57. The method of claim 56, wherein the distributed computing component is an Enterprise JavaBean™ with code that complies with another of the Enterprise JavaBean™ specifications.

58. A method in a data processing system having a distributed computing component with source code, the method comprising the steps of:

receiving an indication to deploy the distributed computing component;

5 retrieving deployment information from a comment of the source code corresponding to the distributed computing component, the deployment information comprising a plurality of properties to control deployment of the distributed computing component on a computer and to control a relationship between the distributed computing component and a deployment environment on the computer;

generating a deployment descriptor file that includes the deployment information;

10 receiving a change to the deployment information associated with the distributed computing component contained in the deployment descriptor file; and

modifying the deployment information in the comment of the source code for the distributed computing component to reflect the change.

59. The method of claim 58, wherein the change reflects a modification of one of the plurality of properties.

60. The method of claim 58, further comprising the step of providing an editor for the deployment descriptor file.

61. The method of claim 58, wherein the editor displays the deployment information in the deployment descriptor file and receives the change to the deployment information.

20 62. The method of claim 61, wherein the editor is an XML editor.

63. A computer-readable medium containing instructions for controlling a data processing system to perform a method, the method comprising the steps of:

receiving a request to generate a distributed computing component;

generating code corresponding to the distributed computing component, the code  
5 containing a method having one of a plurality of types; and

displaying a graphical representation of the code that includes a separately delineated display area for each of the plurality of types.

64. The computer-readable medium of claim 63, wherein the method further comprises the step of displaying a symbol in each separately delineated display area, wherein the  
10 symbol indicates the type of method displayed in the display area.

65. The computer-readable medium of claim 63, wherein the graphical representation of the code includes a separately delineated display area for reference types in the code.

66. The computer-readable medium of claim 63, wherein the one type is a create method type.

67. The computer-readable medium of claim 63, wherein the one type is a finder method  
15 type.

68. The computer-readable medium of claim 63, wherein the one type is a business method type.

69. The computer-readable medium of claim 63, wherein the distributed computing  
20 component is an Enterprise JavaBean™.

70. The computer-readable medium of claim 63, wherein the distributed computing component is a SessionBean.

71. The computer-readable medium of claim 63, wherein the distributed computing component is an EntityBean.

72. A computer-readable medium containing instructions for controlling a data processing system to perform a method, the method comprising the steps of:
  - receiving a request to generate a distributed computing component;
  - generating code corresponding to the distributed computing component, the code
  - 5 containing a method having one of a plurality of types;
  - associating a symbol with each type; and
  - displaying a graphical representation of the method with the symbol indicating the type of the method.
73. The computer-readable medium of claim 72, wherein the step of generating code
- 10 comprises the steps of:
  - generating code corresponding to an implementation class;
  - generating code corresponding to a home interface that is related to the implementation class; and
  - generating code corresponding to a remote interface that is related to the
  - 15 implementation class.
74. The computer-readable medium of claim 73, wherein the code corresponding to the implementation class contains the method and the code corresponding to the home interface contains a signature of the method.
75. The computer-readable medium of claim 74, wherein the method further comprises
- 20 the step of displaying the symbol indicating the type of the method with a graphical representation of the signature.
76. The computer-readable medium of claim 73, wherein the code corresponding to the implementation class contains the method and the code corresponding to the remote interface contains a signature of the method.
77. The computer-readable medium of claim 76, wherein the method further comprises
- 25 the step of displaying the symbol indicating the type of the method with a graphical representation of the signature.

14144582/v2

78. The computer-readable medium of claim 73, wherein the method further comprises the step of displaying a graphical representation of the code corresponding to the implementation class that includes a first separately delineated display area for a create method type, a second separately delineated display area for a finder method type, and a third separately delineated display area for a business method type.

79. The computer-readable medium of claim 73, wherein the method further comprises the step of displaying a graphical representation of the code corresponding to the home interface that includes a separately delineated display area for a create method type and a finder method type.

80. The computer-readable medium of claim 73, wherein the method further comprises the step of displaying a graphical representation of the code corresponding to the remote interface that includes a separately delineated display area for a business method type.

81. A computer-readable medium containing instructions for controlling a data processing system to perform a method, the data processing system comprising a software development tool, the method comprising the steps of:

- initiating execution of the software development tool; and
- 5 while the software development tool is executing, the software development tool performs the steps of:
  - receiving an indication to generate a distributed computing component;
  - generating source code corresponding to the distributed computing component;
  - compiling the source code to form executable code for the distributed computing
  - 10 component;
  - generating a deployment descriptor file that includes a plurality of properties to control deployment of the distributed computing component on a computer and to control a relationship between the distributed computing component and a deployment environment on the computer;
  - 15 storing executable code for the distributed computing component and the deployment descriptor file in a deployment archive;
  - deploying the distributed computing component to the computer; and
  - initiating execution of the distributed computing component on the computer.

82. The computer-readable medium of claim 81, wherein the source code includes an  
20 implementation class.

83. The computer-readable medium of claim 81, wherein the source code includes a home interface.

84. The computer-readable medium of claim 81, wherein the source code includes a remote interface.

25 85. The computer-readable medium of claim 81, wherein the source code includes a primary key class.

86. The computer-readable medium of claim 81, wherein the method further comprises the step of storing deployment information associated with the distributed computing component in a comment of the source code for the distributed computing component.

30 87. The computer-readable medium of claim 86, wherein the step of generating the deployment descriptor file further comprises the steps of:

retrieving the deployment information associated with the distributed computing component from the comment of the source code corresponding to the distributed computing component; and

35 storing the deployment information in the deployment descriptor file.

88. The computer-readable medium of claim 87, wherein the method further comprises the steps of:

receiving a change to the deployment information associated with the distributed computing component contained in the deployment descriptor file; and

40 modifying the deployment information in the comment of the source code for the distributed computing component to reflect the change.

89. The computer-readable medium of claim 86, wherein the deployment information includes at least one of the plurality of properties.

90. The computer-readable medium of claim 81, wherein the step of deploying the distributed computing component comprises the step of transferring the deployment archive to the computer.

91. The computer-readable medium of claim 81, wherein the software development tool is located on the computer.

92. The computer-readable medium of claim 81, wherein the method further comprises the step of initiating execution of a debugger to facilitate testing the distributed computing component on the computer.

93. The computer-readable medium of claim 92, wherein the step of initiating the execution of the distributed computing component comprises initiating the execution of the distributed computing component in one debug session.



55 94. The computer-readable medium of claim 92, wherein the method further comprises the steps of:

receiving an indication of a client software component; and

initiating execution of the client software component in another debug session.

95. The computer-readable medium of claim 94, wherein the client software component  
60 when executing invokes a method associated with the distributed computing component.

96. The computer-readable medium of claim 81, wherein the method further comprises the steps of:

generating a plurality of web pages to test the distributed computing component after deployment; and

65 providing a client application with an address for accessing the web pages.

97. The computer-readable medium of claim 96, wherein one of web pages contains code that is responsive to an invocation from the client application to execute a method in the executable code of the distributed computing component.

98. The computer-readable medium of claim 97, wherein another of the web pages  
70 contains code that provides a result to the client application in response to the execution of the method in the executable code of the distributed computing component.

99. A computer-readable medium containing instructions for controlling a data processing system to perform a method, the data processing system comprising a software development tool, the method comprising the steps of:

initiating the execution of the software development tool; and

5 while the software development tool is executing, the software development tool performing the steps of:

receiving an indication to deploy a distributed computing component;

deploying the distributed computing component to a deployment environment on a computer; and

10 initiating execution of the distributed computing component in debug mode.

100. The computer-readable medium of claim 99, wherein the method further comprises the steps of:

receiving a debug command; and

15 in response to receiving the debug command, controlling the execution of the distributed computing component while in debug mode.

101. The computer-readable medium of claim 100, wherein the method further comprises the step of, in response to receiving the debug command, displaying a result from executing the distributed computing component.

102. The computer-readable medium of claim 99, wherein the step of initiating the execution of the distributed computing component further comprises automatically initiating execution of the distributed computing component in one debug session.

103. The computer-readable medium of claim 102, wherein the method further comprises the steps of:

receiving an indication of a client software component; and

25 automatically initiating execution of the client software component in another debug session.

104. The computer-readable medium of claim 103, wherein the method further comprises the steps of:

receiving a first debug command;

30 receiving a second debug command;

in response to receiving the first debug command, controlling the execution of the client software component in the other debug session; and

in response to receiving the second debug command, controlling the execution of the distributed computing component in the one debug session.

35 105. The computer-readable medium of claim 104, wherein the method further comprises the steps of:

in response to receiving the first debug command, displaying a first result from executing the distributed computing component; and

40 in response to receiving the second debug command, displaying a second result from executing the client software component.

106. The computer-readable medium of claim 105, wherein the client software component when executing invokes a method associated with the distributed computing component.

45 107. The computer-readable medium of claim 106, wherein the first result and the second result reflect the invocation of the method.

108. A computer-readable medium containing instructions for controlling a data processing system to perform a method, the data processing system comprising a software development tool with access to a distributed computing component, the method comprising the steps of:
  - 5 initiating the execution of the software development tool; and  
while the software development tool is executing, the software development tool performing the steps of:
    - receiving an indication to deploy the distributed computing component; and  
generating a web page to facilitate testing the distributed computing component.
- 10 109. The computer-readable medium of claim 108, wherein the method further comprises providing a client with an address for accessing the web page to test the distributed computing component.
- 15 110. The computer-readable medium of claim 109, wherein the web page has code that is responsive to an invocation from the client to execute a method in executable code of the distributed computing component.
111. The computer-readable medium of claim 110, wherein the code of the web page provides a result to the client in response to the execution of the method in the executable code of the distributed computing component.
- 20 112. The computer-readable medium of claim 110, wherein the method corresponds to a create method.
113. The computer-readable medium of claim 110, wherein the method corresponds to a finder method.
114. The computer-readable medium of claim 110, wherein the method corresponds to a business method.

115. A computer-readable medium containing instructions for controlling a data processing system to perform a method, the data processing system having a distributed computing component with code, the distributed computing component having a type, the method comprising the steps of:

5       determining whether the code of the distributed computing component has a non-compliant portion that does not comply with a specification for the type of distributed computing component; and

          when it is determined that the code corresponding to the distributed computing component has a non-compliant portion, replacing the non-compliant portion with new  
10       code that complies with the specification.

116. The computer-readable medium of claim 115, wherein the method further comprises the step of, after replacing the non-compliant portion with new code, refactoring the code corresponding to the distributed computing to modify a reference to the non-complaint portion.

15       117. The computer-readable medium of claim 115, wherein the specification is associated with a target application server that is capable of controlling execution of the distributed computing component.

118. The computer-readable medium of claim 115, wherein the specification corresponds to one of a plurality of Enterprise JavaBean™ specifications.

20       119. The computer-readable medium of claim 118, wherein the distributed computing component is an Enterprise JavaBean™ with code that complies with another of the Enterprise JavaBean™ specifications.

120. A computer-readable medium containing instructions for controlling a data processing system to perform a method, the data processing system having a distributed computing component with source code, the method comprising the steps of:

receiving an indication to deploy the distributed computing component;

5 retrieving deployment information from a comment of the source code corresponding to the distributed computing component, the deployment information comprising a plurality of properties to control deployment of the distributed computing component on a computer and to control a relationship between the distributed computing component and a deployment environment on the computer;

10 generating a deployment descriptor file that includes the deployment information; receiving a change to the deployment information associated with the distributed computing component contained in the deployment descriptor file; and

modifying the deployment information in the comment of the source code for the distributed computing component to reflect the change.

15 121. The computer-readable medium of claim 120, wherein the change reflects a modification of one of the plurality of properties.

122. The computer-readable medium of claim 120, wherein the method further comprises the step of providing an editor for the deployment descriptor file.

20 123. The computer-readable medium of claim 120, wherein the editor displays the deployment information in the deployment descriptor file and receives the change to the deployment information.

124. The computer-readable medium of claim 123, wherein the editor is an XML editor.

125. A data processing system comprising:  
a memory device further comprising a program that receives a request to generate a distributed computing component, that generates code corresponding to the distributed computing component, wherein the code contains a method having one of a plurality of types, and that displays a graphical representation of the code that includes a separately delineated display area for each of the plurality of types; and  
a processor for running the program.
126. The data processing system of claim 125, wherein the program displays a symbol in each separately delineated display area, wherein the symbol indicates the type of method displayed in the display area.
127. The data processing system of claim 125, wherein the graphical representation of the code includes a separately delineated display area for reference types in the code.
128. The data processing system of claim 125, wherein the one type is a create method type.
129. The data processing system of claim 125, wherein the one type is a finder method type.
130. The data processing system of claim 125, wherein the one type is a business method type.
131. The data processing system of claim 125, wherein the distributed computing component is an Enterprise JavaBean™.
132. The data processing system of claim 125, wherein the distributed computing component is a SessionBean.
133. The data processing system of claim 125, wherein the distributed computing component is a EntityBean.

134. A data processing system comprising:

a memory device further comprising a program that receives a request to generate a distributed computing component, that generates code corresponding to the distributed computing component, wherein the code contains a method having one of a plurality of types, that associates a symbol with each type, and that displays a graphical representation of the method with the symbol indicating the type of the method; and

5 a processor for running the program.

135. The data processing system of claim 134, wherein when the program generates code corresponding to the distributed computing component, the program generates code corresponding to an implementation class, generates code corresponding to a home interface that is related to the implementation class, and generates code corresponding to a remote interface that is related to the implementation class.

136. The data processing system of claim 135, wherein the code corresponding to the  
implementation class contains the method and the code corresponding to the home  
15 interface contains a signature of the method.

137. The data processing system of claim 136, wherein when the program displays the symbol indicating the type of the method with a graphical representation of the signature.

138. The data processing system of claim 135, wherein the code corresponding to the  
implementation class contains the method and the code corresponding to the remote  
20 interface contains a signature of the method.

139. The data processing system of claim 138, wherein when the program the symbol indicating the type of the method with a graphical representation of the signature.

140. The data processing system of claim 135, wherein the program displays a graphical representation of the code corresponding to the implementation class that includes a first separately delineated display area for a create method type, a second separately delineated display area for a finder method type, and a third separately delineated display area for a business method type.



141. The data processing system of claim 135, wherein when the program displays a graphical representation of the code corresponding to the home interface that includes a  
30 separately delineated display area for a create method type and a finder method type.

142. The data processing system of claim 135, wherein when the program displays a graphical representation of the code corresponding to the remote interface that includes a separately delineated display area for a business method type.

14144582/v2

143. A data processing system comprising:

a memory device further comprising a program that receives an indication to generate a distributed computing component, that generates source code corresponding to the distributed computing component, that compiles the source code to form executable code for the distributed computing component, that generates a deployment descriptor file that includes a plurality of properties to control deployment of the distributed computing component on a computer and to control a relationship between the distributed computing component and a deployment environment on the computer, that stores the executable code for the distributed computing component and the deployment descriptor file in a deployment archive, that deploys the distributed computing component to the computer, and that initiates execution of the distributed computing component on the computer; and  
a processor for running the program.

144. The data processing system of claim 143, wherein the source code includes an implementation class.

145. The data processing system of claim 143, wherein the source code includes a home interface.

146. The data processing system of claim 143, wherein the source code includes a remote interface.

147. The data processing system of claim 143, wherein the source code includes a primary key class.

148. The data processing system of claim 143, wherein the program further stores deployment information associated with the distributed computing component in a comment of the source code for the distributed computing component.

149. The data processing system of claim 148, wherein when the program generates the deployment descriptor file, the program retrieves the deployment information associated with the distributed computing component from the comment of the source

code corresponding to the distributed computing component, and stores the deployment information in the deployment descriptor file.

30 150. The data processing system of claim 149, wherein the program further receives a change to the deployment information associated with the distributed computing component contained in the deployment descriptor file, and modifies the deployment information in the comment of the source code for the distributed computing component to reflect the change.

35 151. The data processing system of claim 149, wherein the deployment information includes at least one of the plurality of properties.

152. The data processing system of claim 143, wherein when the program deploys the distributed computing component, the program transfers the deployment archive to the computer.

40 153. The data processing system of claim 143, wherein the software development tool is located on the computer.

154. The data processing system of claim 143, wherein the program initiates execution of a debugger to facilitate testing the distributed computing component on the computer.

45 155. The data processing system of claim 154, wherein when the program initiates the execution of the distributed computing component, the program initiates the execution of the distributed computing component in one debug session.

156. The data processing system of claim 154, wherein the program receives an indication of a client software component, and initiates execution of the client software component in another debug session.

50 157. The data processing system of claim 156, wherein the client software component when executing invokes a method associated with the distributed computing component.

158. The data processing system of claim 143, wherein the program further generates a plurality of web pages to test the distributed computing component after deployment, and provides a client application with an address for accessing the web pages.

55 159. The data processing system of claim 158, wherein one of the web pages contains code that is responsive to an invocation from the client application to execute a method in the executable code of the distributed computing component.

160. The data processing system of claim 159, wherein another of the web pages contains code that provides a result to the client application in response to the execution  
60 of the method in the executable code of the distributed computing component.

161. A system comprising:

means for receiving a request to generate a distributed computing component;

means for generating code corresponding to the distributed computing component, the code containing methods of a plurality of types; and

5 means for displaying a graphical representation of the code that includes a separately delineated display area for one of the plurality of types.